Neural Network Optimization for Combinations in Identification Systems

Sergey Tulyakov and Venu Govindaraju

Center for Unified Biometrics and Sensors,
University at Buffalo, Buffalo, USA
{tulyakov,venu}@cubs.buffalo.edu
http://www.cubs.buffalo.edu/

Abstract. In this paper we investigate the construction of combination functions in identification systems. In contrast to verification systems, the optimal combination functions for identification systems are not known. In this paper we represent the combination function by means of a neural network and explore different methods of its training, so that the identification system performance is optimized. The modifications are based on the principle of utilizing best impostors from each training identification trial. The experiments are performed on score sets of biometric matchers and handwritten word recognizers. The proposed combination methods are able to outperform the likelihood ratio, which is optimal combination method for verification system, as well as, weighted sum combination method optimized for best performance in identification systems.

Keywords: Classifier combination, identification system, biometric matchers, neural network.

1 Introduction

Suppose we have a matching system with N registered or enrolled classes. Given some input, the system has to find its best match to any of the N enrolled classes. We call such matching system an *identification system*. In contrast to more general N-class pattern classification problem setup, we imply that the matching score of the input to the enrolled class is derived using only input and enrolled templates. Consequently, identification systems can deal with variable and large number of classes N and no retraining of the matching algorithm is needed. The examples of identification systems include biometric identification systems and handwritten word recognition; both can contain large and variable number of classes, persons or lexicon words, and matching algorithms, as a rule, calculate the matching score using only two, input and enrolled, templates.

Identification systems are different from verification systems. In *verification systems* the possible class of the input is provided beforehand; the system only performs the match of the input to the enrolled template of the specified class, and depending on the matching score outputs accept or reject decision. In *identification system* we have to match the input against all enrolled templates and output the class matching the input best. For evaluating performance of verification systems we can use ROC curves, and for evaluating performance of

J.A. Benediktsson, J. Kittler, and F. Roli (Eds): MCS 2009, LNCS 5519, pp. 418–427, 2009.

identification systems we can calculate the correct identification rate or use rank measures such as cumulative match curves (CMC).

In this paper we consider the combinations of matching scores in identification systems. Though we used only pairs of matchers for combination, the presented algorithms can be applied to situations with few matchers. Each matcher $j = 1, 2$ produces sets of matching scores s_i^j assigned to each of $i = 1, \ldots, N$ classes. A combination function f is used to combine 2 matching scores corresponding to each class: $S_i = f(s_i^1, s_i^2)$. In identification systems the classification result C is determined as

$$C = \arg \max_{i=1,\ldots,N} f(s_i^1, s_i^2) \tag{1}$$

In verification systems, on the other hand, we compare the value of combination function to some threshold and make accept/reject decision.

The optimal combination function for verification system is well-known [1,2]; such function should have decision surfaces separating two types of score pairs (s_i^1, s_i^2), genuine and impostor. Optimal Bayes classifier separating genuine and impostor score pairs is obtained by the ratio of genuine and impostor score densities (*likelihood ratio*):

$$f_{lr}(s_i^1, s_i^2) = \frac{p_{gen}(s_i^1, s_i^2)}{p_{imp}(s_i^1, s_i^2)} \tag{2}$$

and can be well approximated if the number of matchers (two in our case) is relatively small.

On the other hand, the solution to finding optimal combination function for identification system is not yet known. As we showed in [2], likelihood ratio f_{lr} is optimal if matching scores assigned to different classes are statistically independent. If they are dependent, likelihood ratio might be non-optimal, and the performance of combined system can be even worse than the performance of a single matcher. The scores assigned to different classes are usually dependent since they are derived using same input template.

The goal of this paper is to investigate the approaches of constructing combination function for identification systems with the help of multi-layer perceptron. In particular, we present two methods of training neural network resulting in increased identification system performance.

2 Previous Work

Previous work in classifier combinations and combinations of biometric matchers makes little distinction on whether the considered system is verification or identification system. For example, Kittler et al. [3] derive the combination rules assuming identification system, but test them using verification system. Besides, the independence of matching scores assigned to different classes is assumed in that work.

As another example, Lee et al. [4] explicitly reduce the problem of combining matchers in a biometric identification system to the task of applying a classifier (SVM) trained for an equivalent verification system. Since the combination function is trained for verification system, it may not produce an optimal combination algorithm for identification systems.

Whereas most research in combinations of biometric matchers deals with verification systems, e.g. [5], the earlier research in classifier combinations dealt with more general classifiers. Effectively, many of the earlier approaches were learning combination functions of the following type:

$$C = \arg \max_{i=1,\dots,N} f_i(\{s_k^1\}_{k=1,\dots,N}, \{s_k^2\}_{k=1,\dots,N}) \tag{3}$$

instead of less complex combinations of Eq. 1. For example, Bayesian and Dempster-Shafer combination methods of [6] require learning confusion matrices for each classifier participating in the combination. The Behavior-Knowledge Space combination method of [7] requires learning a decision space of a set of classifiers participating in the combination. Although these approaches can be considered to be somewhat optimal, they could be applied only in situations with a small number of classes. However, in our applications of biometrics and handwritten word recognition, the number of classes N is of the order of thousands and we are forced to construct combinations of the Eq. 1 type.

Some of the earlier works on the training of pattern recognition systems recognized the need to train the algorithms with the goal of minimizing the classification errors. As noted in [8], the traditional neural network training involving MSE (mean squared error) minimization might not result in the neural network having minimum classification errors. Different methods of training neural networks for classification error minimization have been proposed [8,9,10]. Though our application of neural network used as combination function of Eq. 1 is different from the application of neural network as pattern classifiers in these previous works, which are rather similar to Eq. 3, we employ a training principle similar to principle proposed in those works - we will be utilizing training samples proved to be most difficult for classification.

We have previously underscored the need to have a separate training procedures of combination algorithms in verification and identification systems [2]. Furthermore, we proposed some heuristic methods of constructing combination functions for identification systems in [11]. In this paper we are looking for the ways to change the training procedures of traditional multilayer perceptron neural networks, so that the resulting combination function has optimized performance in identification systems. Methods considered in this paper can be viewed as a more automated and generalized compared to the heuristic methods described in [11].

3 Optimizing Combination Functions for Identification Systems

3.1 Weighted Sum Combination

One of the most frequently used methods for combining matching scores in identification systems is the weighted sum rule. In our case, we combine only two matchers and the weighted sum combination function can be written as

$$f(s^1, s^2) = ws^1 + (1 - w)s^2 \tag{4}$$

The weight w can be chosen heuristically so that the better performing matchers have a bigger weight [5]. The optimal weights can be also estimated for linear combinations of classifiers subject to the minimization of classification error [12].

In our experiments we have trained the weights so that the number of successful identification trials on the training set is maximized. The previously proposed methods of training resulting in the minimization of classification error [12] are not directly applicable due to much bigger number of classes in our case. Since we have only two matchers in all our configurations, it was possible to utilize a brute-force approach: we calculate the correct identification rate of the combination function $f(s^1, s^2) = ws^1 + (1 - w)s^2$ for different values of $w \in [0, 1]$, and find w corresponding to the highest recognition rate. Despite being brute-force, due to simplicity of weighted sum method, this approach was the fastest to train.

3.2 Minimizing Classification Error and Iterative Learning

If we perform training for verification system, we can treat genuine and impostor scores from different identification trials separately. Indeed, the optimal combination in the form of likelihood ratio (Eq. 2) uses separately approximated genuine and impostor densities, and any other algorithm can do the same. But in order to perform training of combination function for identification system, we have to consider the scores in each identification trial as a single training sample, and train the combination function on these samples. This is precisely the technique used to train the weighted sum rule for identification systems (Section 3.1). For each training identification trial we check whether the genuine score pair produced greater combined scores than all the impostor score pairs. By counting the numbers of successful trials we were able to choose the proper weights. Although the weighted sum rule provides a reasonable performance in our applications, its decision surfaces are linear and might not completely separate the generally non-linear score distributions. Therefore we explore more complex combination functions trained with the available training set.

In this paper we explore the approximation of combination function $f(s^1, s^2)$ by means of neural network, multilayer perceptron. Although the previous work in neural network optimization for minimizing misclassification errors was in constructing classifiers and not their combinations [8,9,10], we apply similar optimization criteria for the training. In [9] several optimization criteria were explored. The general solution consists in constructing a smooth misclassification cost function giving different weights to different errors of currently trained neural network, and modifying neural network by gradient descent method to reduce the cost. In our case, we used one particular case of such cost - the cost incurred by the largest possible error from the best impostor. Such cost is an extreme case of parametric cost functions considered in [9], where the parameter is chosen so that the cost function uses only best impostor.

Another difference of our approach with previous research in minimum classifier error optimization of neural networks is that during network update the input to our neural network can consist of only one score pair (this is difference between Eq. 3 and Eq. 1). As a consequence, we are not only required to modify the cost function determined by network outputs, but we also need to provide a proper score pair as the training input. By considering only the best impostor score pair we are able to do it.

In order to implement our algorithms, we need to be able to determine what is the best impostor score pair in each training identification trial. The best impostor depends on the currently trained combination function. Therefore, for our methods we use the following iterative training procedure:

1. Make initialization of $f(s^1, s^2)$.
2. For each training identification trial find the impostor score pair with the biggest value of the combined score according to currently trained $f(s^1, s^2)$.
3. Update $f(s^1, s^2)$ by using genuine score pair and found best impostor score pair of one identification trial.
4. Repeat steps 2-3 for all training identification trials.
5. Repeat steps 2-4 for predetermined number of training epochs.

Note, that proposed training procedure based on best impostors does not explicitly model the dependence between matching scores assigned to different classes. Though such modeling can be very helpful for improving the performance of combination algorithms, it leads to a different type of combinations not defined by Eq. 1 [13]. In current paper we are interested in modified optimization criteria (minimizing classification error), and by using this criteria we implicitly account for dependencies between matching scores assigned to different classes.

3.3 Neural Network Training for Identification Systems

Neural networks, especially multilayer perceptrons, allow approximation of arbitrary functions. Therefore, it should be possible to train a neural network to represent the optimal combination function in identification systems. This approach can be viewed as a generalization of the combination method using a sum of logistic functions described in our previous work [11]. A neural network with logistic activation functions and a single layer of hidden nodes directly corresponds to the combination function consisting of the sum of logistic functions. By utilizing more than one hidden layer and by using generic training of the neural network, it is possible to obtain better combination function than by using ad hoc structure and training procedure of the sum of logistic functions [11].

We compare three approaches for training a neural network for the combination task at hand. The first approach is the traditional training using separate genuine and impostor scores. The other two approaches focus on minimizing the misclassification rate, and the genuine and impostor scores are not treated separately.

1. Traditional training: random impostor score pairs are used alongside with genuine score pairs.

2. Best impostor training: following iterative training procedure, the best impostor score pair is found from an identification trial and used together with genuine score pair to update neural network.
3. Mixed scores training: we use best impostor from the identification trial for training only if there is a failure in one combined classifier. More precisely, let (s^1_{gen}, s^2_{gen}) and (s^1_{bi}, s^2_{bi}) denote the genuine and best impostor score pairs for the current identification trial. We update the neural network only if $s^1_{gen} > s^1_{bi}$ and $s^2_{gen} < s^2_{bi})$ or $(s^1_{gen} < s^1_{bi}$ and $s^2_{gen} > s^2_{bi})$. This training method can be viewed as a combination of best impostor neural network training and the conditional training of the sum of logistic functions combination method [11].

Our goal is to train the neural network so that the misclassification rate is minimized. As we discussed in section 3.2, the configuration of our neural network is different from the networks trained with the purpose of classifier error minimization. But we can notice that used optimization criteria are similar. Indeed, by considering the best impostor we effectively use the extreme case of parametric cost functions presented in [9]. During our training of neural network we employ the mean square error defined for genuine and best impostor samples; such choice of error calculation corresponds to considering the square polynomial functions for cost calculations in [9]. The mixed score training implies additional selection of training samples, and can also be represented by a proper choice of cost function family.

4 Experiments

4.1 Handwritten Word Recognizers

We consider the application of handwritten word recognizers in the automatic processing of United Kingdom mail. The destination information of the mail piece contains the name of the postal town or county. After automatic segmentation of the mail piece image, the goal of the handwritten word recognizer is to match the hypothesized town or county word image against a lexicon of possible names, which contains 1681 entries.

We use two handwritten word recognizers for this application: Character Model Recognizer (CMR)[14] and Word Model Recognizer (WMR)[15]. Both recognizers employ similar approaches to word recognition: they oversegment the word images, match the combinations of segments to characters and derive a final matching score for each lexicon word as a function of the character matching scores.

Our data consists of three sets of word images of approximately the same quality. The data was initially provided as these three subsets and therefore we did not regroup them. The images were manually truthed and only those images containing any of the 1681 lexicon words were retained. The word recognizers were run on these images and their match scores for all 1681 lexicon words were saved. Note, that both recognizers reject some lexicon entries if, for example, the lexicon word is too short or too lengthy for the presented image. We assume that in real systems such rejects will be dealt with separately (it is possible that the lexicon word corresponding to image truth will be rejected), but for our combination experiments we keep only the scores of those lexicon words which are not rejected by either of the recognizers. Thus for each image I_k we have a variable number N_k of score pairs (s_i^{cmr}, s_i^{wmr}), $i = 1, \ldots, N_k$ corresponding to non-rejected lexicon words. One of these pairs corresponds to the true word of the image which we refer to as 'genuine' scores, and the other 'impostor' score pairs correspond to non-truth words.

After discarding images with non-lexicon words, and images where the truth word was rejected by either recognizer, we are left with three sets of 2654, 1723 and 1770 images and related sets of score pairs. We will refer to the attempt of recognizing a word image as an identification trial. Thus each identification trial has a set of score pairs (s_i^{cmr}, s_i^{wmr}), $i = 1, \ldots, N_k$ with one genuine score pair and $N_k - 1$ impostor pairs. The scores of each recognizer were also linearly normalized so that each score is in the interval $[0, 1]$ and bigger score implies a better match.

Since our data was already separated into three subsets, we used this structure for producing the training and testing sets. Each experiment was repeated three times. Each time one subset is used as a training set, and the other two sets are used as test sets. The final results are derived as averages of these three training/testing phases.

4.2 Biometric Person Matchers

We used biometric matching score set BSSR1 distributed by NIST[16]. This set contains matching scores for a fingerprint matcher and two face matchers 'C' and 'G'. Fingerprint matching scores are given for left index 'li' finger matches and right index 'ri' finger matches. For our experiments we used four combinations involving both fingerprint and face score subsets: 'li&C', 'li&G', 'ri&C' and 'ri&G'

Though the BSSR1 score set has a subset of scores obtained from the same physical individuals, this subset is rather small - 517 identification trials with 517 enrolled persons. Therefore we used larger subsets of fingerprint and face matching scores of BSSR1 by creating virtual persons. The fingerprint scores of a virtual person come from a physical person and the face scores come from a different individual. The scores are not reused, and thus we are limited to a maximum of 6000 identification trials and a maximum of 3000 classes (or enrolled persons). Some enrollees and some identification trials also needed to be discarded since the corresponding matching scores were invalid probably due to enrollment errors. Finally, we split the data into two parts - 2991 identification trials with 2997 enrolled persons, with each part used as training and testing sets in two phases. The final results are the averages of these two phases.

4.3 Experimental Results

In the likelihood ratio method we reconstructed the densities using the Parzen window method with Gaussian kernels. The window widths are found by maximum likelihood leave-one-out cross validation method on a training set. Note that the reconstructed densities $p_{gen}(s^1, s^2)$ and $p_{imp}(s^1, s^2)$ of the likelihood ratio combination function 2 are two-dimensional. Given a large number of training samples, using two-dimensional kernels in the Parzen method results in a good approximation of the densities [17].

For the weighted sum combination method 4, as well for other methods, we use separate training and testing subsets. It is worth noting, that despite of only single weight w to be found, weighted sum method indeed has a slightly lower performance on the testing sets than on the training set.

In all the cases of the neural network methods we have the same configuration - multilayer perceptron with configuration 2-8-9-1, sigmoid activation functions and backpropagation training. We keep the default parameter settings of the neural network library [18]. About 100 training epochs are required to get the best performance with minimal overfitting effect. Since we did not have a separate validation set for the biometric dataset, we decided to run the training for 300 epochs and choose the best performance numbers on test datasets.

Table 1. The results of experiments. Numbers represent the correct identification rates (in %).

Matchers	Likelihood Ratio	Weighted Sum Rule	Traditional Training	Best Impostor Training	Mixed Scores Training
CMR&WMR	69.84	81.58	76.69	80.54	**81.67**
li&C	97.24	97.23	97.01	97.26	**97.39**
li&G	95.90	95.47	96.00	96.07	**96.29**
ri&C	98.23	98.09	98.21	98.26	**98.33**
ri&G	97.14	96.82	97.41	**97.43**	97.38

The results of the experiments are presented in Table 1. The numbers in the table refer to the correct identification rates, that is the percentage of trials in which the genuine score receives the best score compared to impostor scores. In general, neural networks showed slightly better results which can be explained by their superior trainability compared to density based likelihood ratio method and linear weighted sum method.

As we discussed in [2], the likelihood ratio method actually fails for combination of word recognizers - it has lower performance than WMR alone. Such result is explained by the strong dependence between WMR's matching scores assigned to different classes. But likelihood ratio has slightly better performance than weighted sum for combination of biometric matchers due to weaker dependence between scores and the inability of weighted sum to model non-linear decision boundaries. The goal of considered neural network combination is to be able to outperform both likelihood ratio and weighted sum. As we can see from Table 1, our modifications to neural network training achieved this task. The last modification, mixed scores training, is able to outperform the likelihood ratio weighted sum combination in all cases.

5 Summary

Verification and identification systems possess different optimal combination functions, and therefore require different training procedures. The optimal combination function for verification systems coincides with the likelihood ratio of genuine and impostors scores. We can approximate this function directly by reconstructing score densities, as we did in this paper, or use traditional pattern classification algorithms trained to separate genuine and impostor scores. The optimal combination function for identification system, on the other hand, is difficult to find.

In this paper we investigated the approaches of constructing combination functions for identification systems by means of neural networks. Previous works in neural network optimizations suggest the possibility that our optimization modifications might have a property of optimality. The experiments on biometric matchers and handwritten word recognizers show that proposed methods are able to outperform likelihood ratio, as well as traditionally used in identification system, weighted sum combination method.

References

1. Prabhakar, S., Jain, A.K.: Decision-level fusion in fingerprint verification. Pattern Recognition 35(4), 861–874 (2002)
2. Tulyakov, S., Govindaraju, V., Wu, C.: Optimal classifier combination rules for verification and identification systems. In: Haindl, M., Kittler, J., Roli, F. (eds.) MCS 2007. LNCS, vol. 4472, pp. 387–396. Springer, Heidelberg (2007)
3. Kittler, J., Hatef, M., Duin, R., Matas, J.: On combining classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence, 226–239 (March 1998)
4. Lee, Y., Lee, K., Jee, H., Gil, Y., Choi, W., Ahn, D., Pan, S.: Fusion for multimodal biometric identification. In: Kanade, T., Jain, A., Ratha, N.K. (eds.) AVBPA 2005. LNCS, vol. 3546, pp. 1071–1079. Springer, Heidelberg (2005)
5. Snelick, R., Uludag, U., Mink, A., Indovina, M., Jain, A.: Large-scale evaluation of multimodal biometric authentication using state-of-the-art systems. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(3), 450–455 (2005)
6. Xu, L., Krzyzak, A., Suen, C.Y.: Methods for combining multiple classifiers and their applications to handwriting recognition. IEEE transactions on System, Man, and Cybernetics 23(3), 418–435 (1992)
7. Huang, Y., Suen, C.: A method of combining multiple experts for the recognition of unconstrained handwritten numerals. IEEE Transactions on Pattern Analysis and Machine Intelligence 17(1), 90–94 (1995)
8. Hampshire II, J.B., Waibel, A.H.: A novel objective function for improved phoneme recognition using time-delay neural networks. IEEE Transactions on Neural Networks 1(2), 216–228 (1990)
9. Juang, B.H., Katagiri, S.: Discriminative learning for minimum error classification [pattern recognition]. IEEE Transactions on Signal Processing, IEEE Transactions on see also Acoustics, Speech, and Signal Processing 40(12), 3043–3054 (1992)
10. Nedeljkovic, V.: A novel multilayer neural networks training algorithm that minimizes the probability of classification error. IEEE Transactions on Neural Networks 4(4), 650–659 (1993)

11. Tulyakov, S., Wu, C., Govindaraju, V.: Iterative methods for searching optimal classifier combination function. In: First IEEE International Conference on Biometrics: Theory, Applications, and Systems, 2007. BTAS 2007, pp. 1–5 (2007)
12. Ueda, N.: Optimal linear combination of neural networks for improving classification performance. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(2), 207–215 (2000)
13. Tulyakov, S., Govindaraju, V.: Use of identification trial statistics for combination of biometric matchers. IEEE Transactions on Information Forensics and Security 3(4), 719–733 (2008)
14. Favata, J.: Character model word recognition. In: Fifth International Workshop on Frontiers in Handwriting Recognition, Essex, England, pp. 437–440 (1996)
15. Kim, G., Govindaraju, V.: A lexicon driven approach to handwritten word recognition for real-time applications. IEEE Transactions on Pattern Analysis and Machine Intelligence 19(4), 366–379 (1997)
16. Nist biometric scores set, `http://www.nist.gov/biometricscores/`
17. Tulyakov, S., Govindaraju, V.: Utilizing independence of multimodal biometric matchers. In: Gunsel, B., Jain, A.K., Tekalp, A.M., Sankur, B. (eds.) MRCS 2006. LNCS, vol. 4105, pp. 34–41. Springer, Heidelberg (2006)
18. Nissen, S.: Implementation of a fast artificial neural network library (fann). Technical report, Department of Computer Science, University of Copenhagen (2003)